2611 #2 RT 10-25-01

Docket #: SVL920010039US1 PATENT

OIPE JC24 JUL 09 2001 PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Shyh-Mei F. Ho, et al.**

Application No.: **09/849,816** Group No.: **Unknown**

Filed: **May 4, 2001** Examiner: **Unknown**

Title: **IMS TRANSACTION MESSAGES METAMODEL**

RECEIVED JUL 12 2001 TC 2600 MAILROOM

LETTER TRANSMITTING FORMAL DRAWINGS

Commissioner for Patents
Washington, D.C. 20231

Attn: Official Draftsman

Dear Sir:

Attached please find eight sheets (8) of formal patent drawings for above-referenced application which was filed with informal drawings on May 4, 2001.

RECEIVED JUL 17 2001 Technology Center 2100

Respectfully Submitted,

CERTIFICATE OF MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the Commissioner for Patents, Washington, D.C. 20231.

Date of deposit: July 5, 2001

Person mailing paper: Carla R. Aliberti

Signature: Carla R. Aliberti

Richard M. Goldman
Registration No: 25,585
Attorney for Applicant
Telephone (408) 463-5891
IBM Corporation
Intellectual Property Law (J46/G4)
555 Bailey Avenue
San Jose, CA 95141-1003

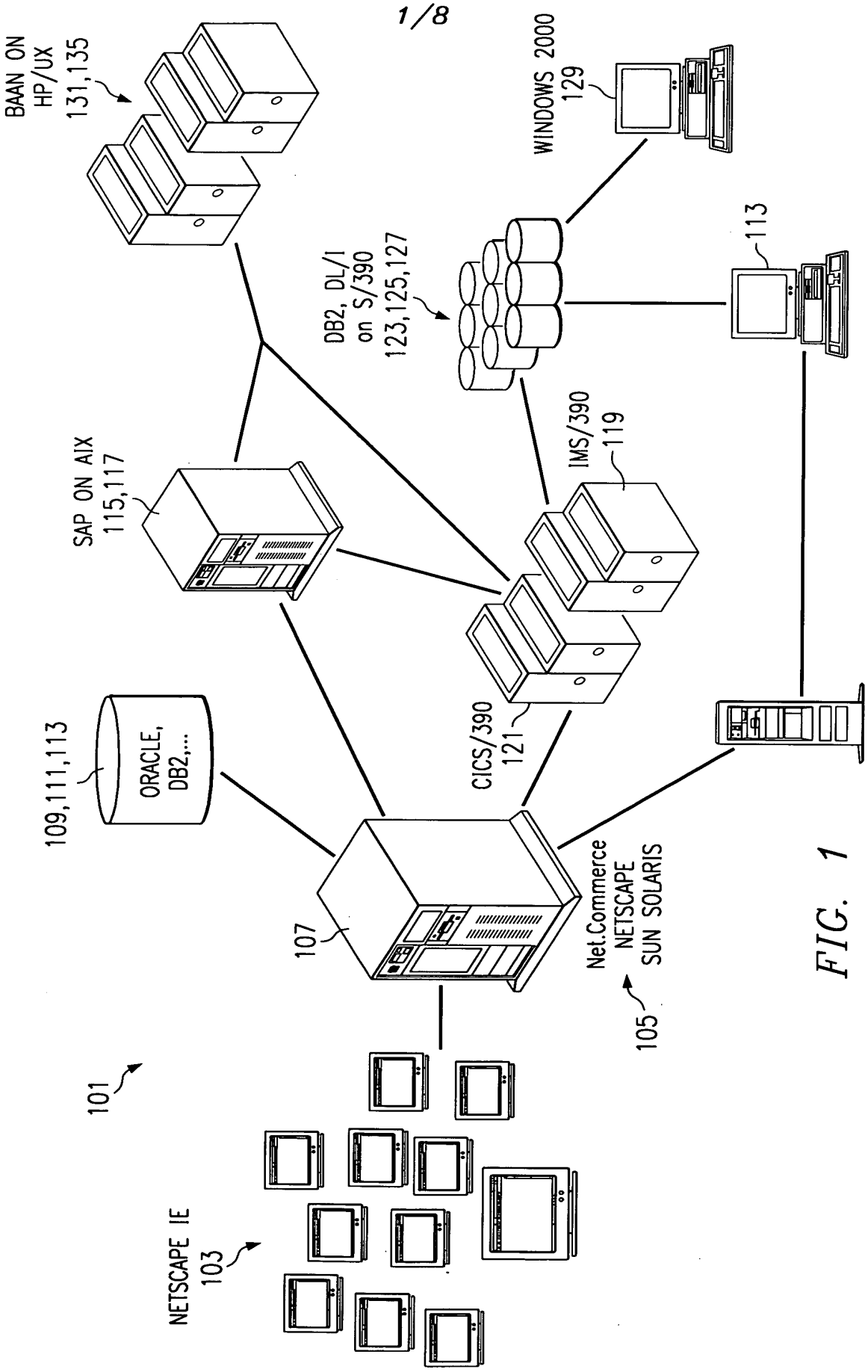
BAAN ON HP/UX 131,135
WINDOWS 2000 129
DB2, DL/I on S/390 123,125,127
113
SAP ON AIX 115,117
IMS/390 119
109,111,113
ORACLE, DB2,...
CICS/390 121
107
Net.Commerce NETSCAPE SUN SOLARIS
105
101
NETSCAPE IE 103

FIG. 1

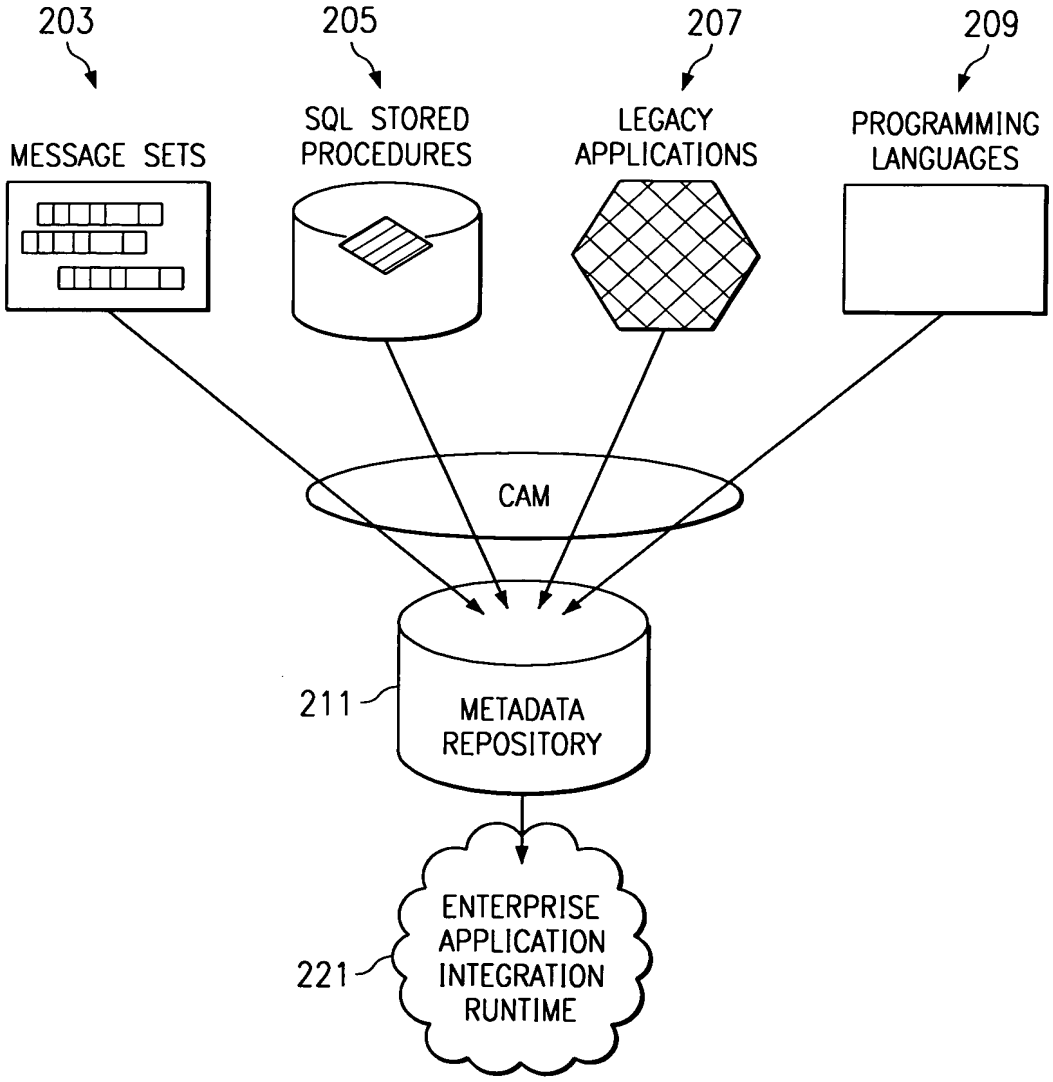
203
205
207
209
MESSAGE SETS
SQL STORED PROCEDURES
LEGACY APPLICATIONS
PROGRAMMING LANGUAGES
CAM
211
METADATA REPOSITORY
221
ENTERPRISE APPLICATION INTEGRATION RUNTIME

FIG. 2

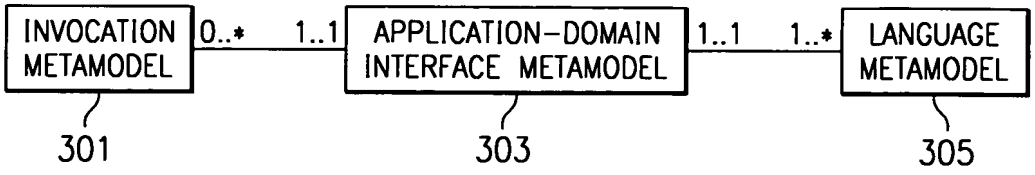
INVOCATION METAMODEL
0..*
1..1
APPLICATION-DOMAIN INTERFACE METAMODEL
1..1
1..*
LANGUAGE METAMODEL
301
303
305

FIG. 3

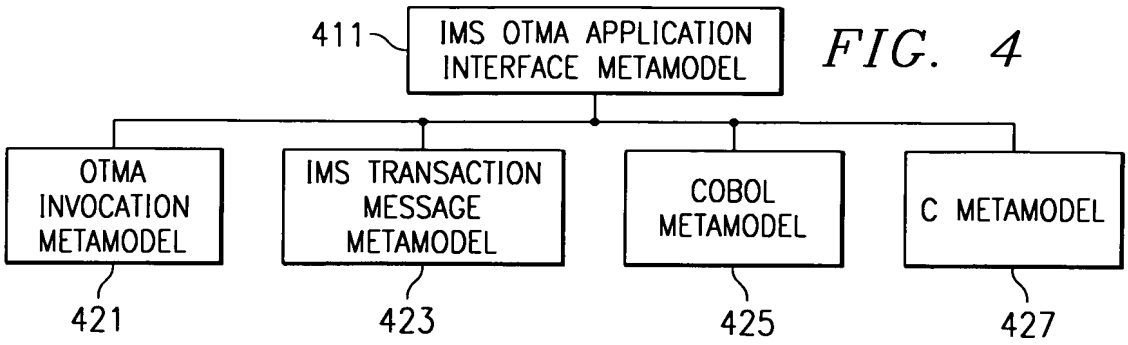

FIG. 4

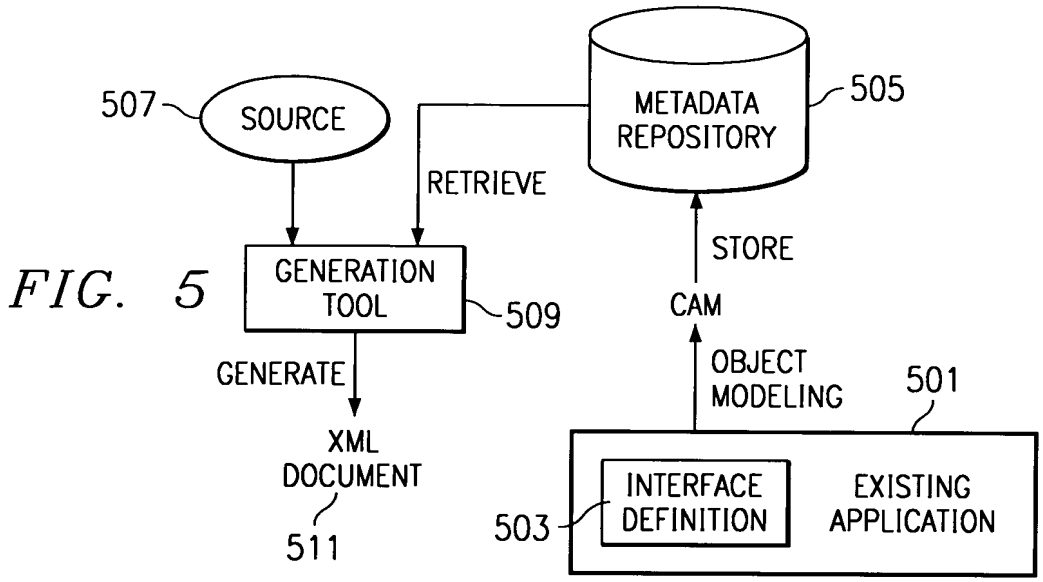

FIG. 5

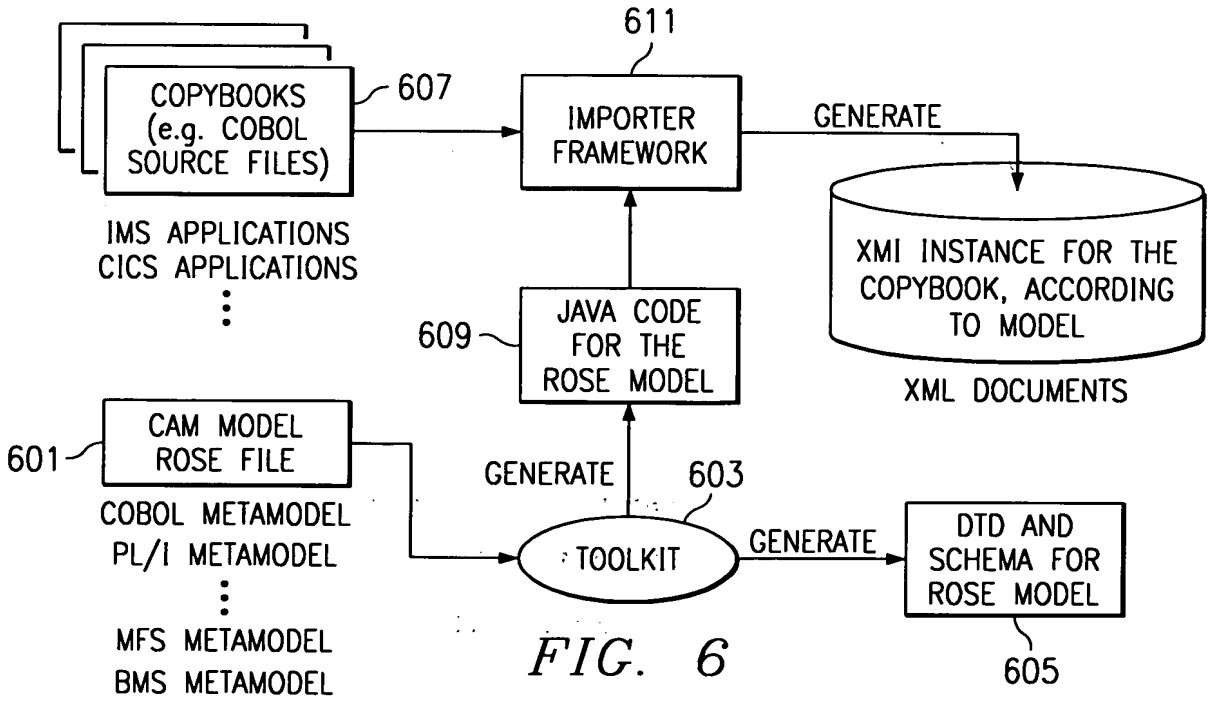

FIG. 6

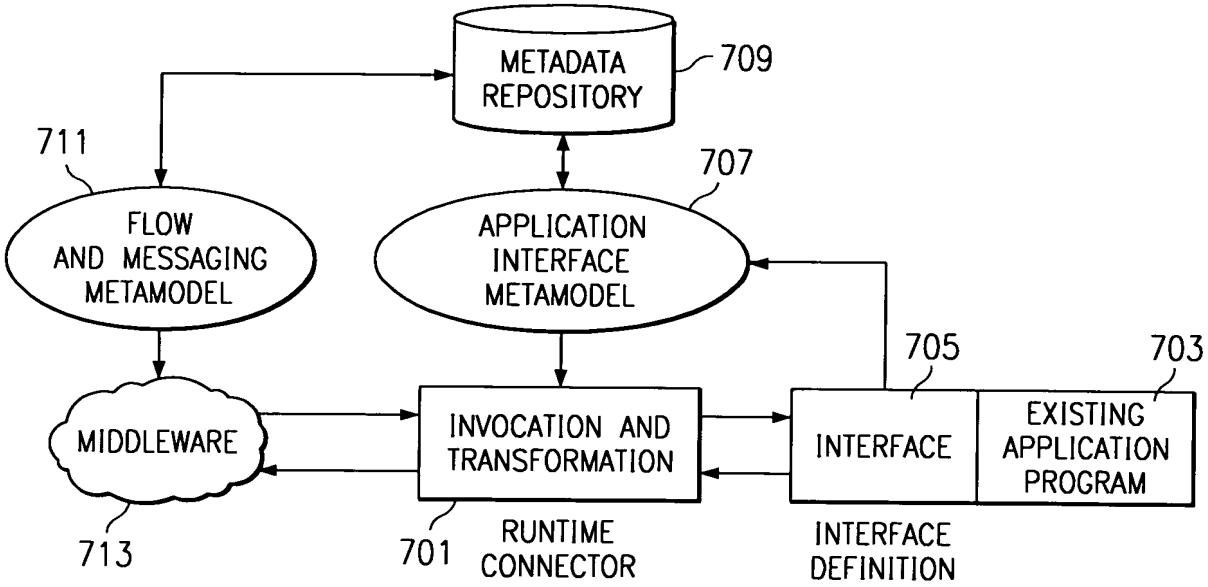
METADATA REPOSITORY
709
711
707
FLOW AND MESSAGING METAMODEL
APPLICATION INTERFACE METAMODEL
705
703
MIDDLEWARE
INVOCATION AND TRANSFORMATION
INTERFACE
EXISTING APPLICATION PROGRAM
713
701
RUNTIME CONNECTOR
INTERFACE DEFINITION

FIG. 7

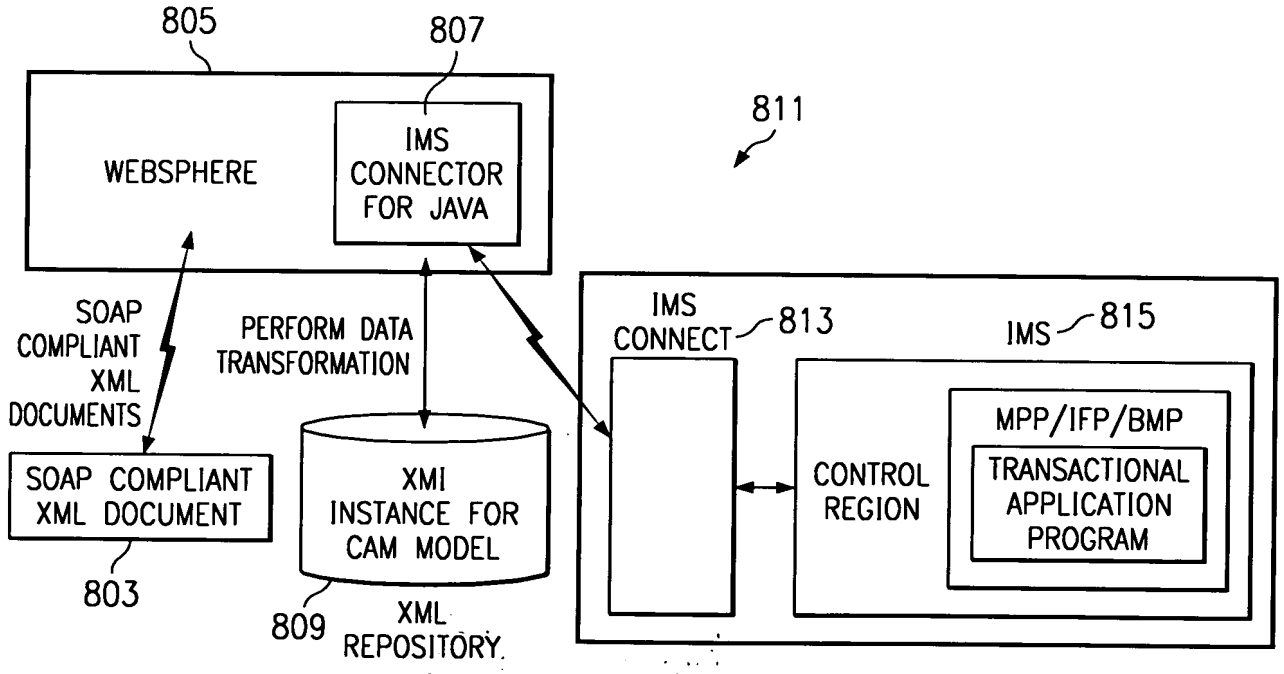
805
807
811
WEBSPHERE
IMS CONNECTOR FOR JAVA
SOAP COMPLIANT XML DOCUMENTS
PERFORM DATA TRANSFORMATION
IMS CONNECT
813
IMS
815
MPP/IFP/BMP
TRANSACTIONAL APPLICATION PROGRAM
CONTROL REGION
SOAP COMPLIANT XML DOCUMENT
XMI INSTANCE FOR CAM MODEL
803
809
XML REPOSITORY

FIG. 8

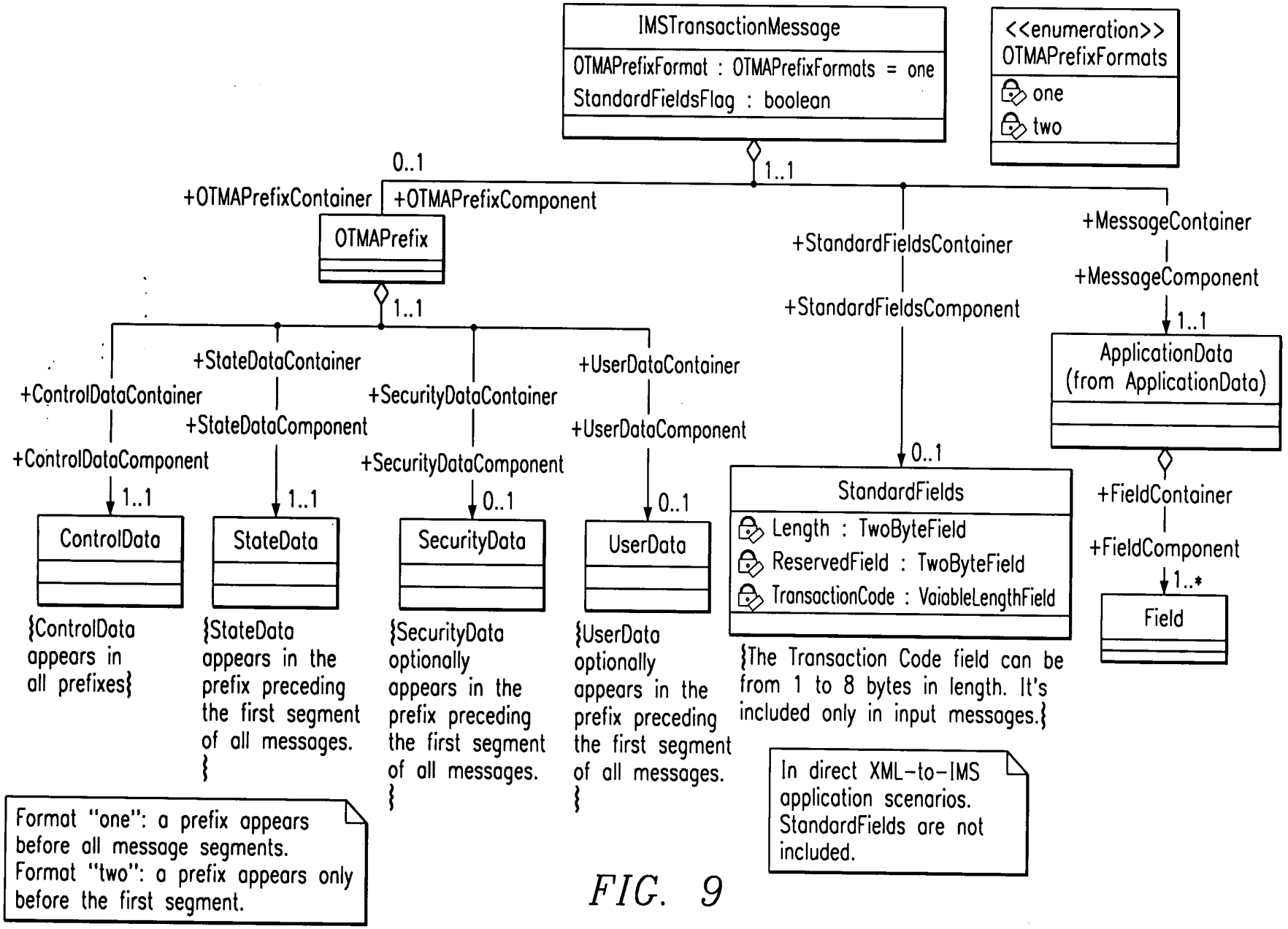
IMSTransactionMessage
OTMAPrefixFormat : OTMAPrefixFormats = one
StandardFieldsFlag : boolean
<<enumeration>>
OTMAPrefixFormats
one
two
0..1
1..1
+OTMAPrefixContainer
+OTMAPrefixComponent
OTMAPrefix
1..1
+StandardFieldsContainer
+StandardFieldsComponent
+MessageContainer
+MessageComponent
1..1
ApplicationData
(from ApplicationData)
+FieldContainer
+FieldComponent
1..*
Field
+ControlDataContainer
+ControlDataComponent
1..1
ControlData
{ControlData appears in all prefixes}
+StateDataContainer
+StateDataComponent
1..1
StateData
{StateData appears in the prefix preceding the first segment of all messages.}
+SecurityDataContainer
+SecurityDataComponent
0..1
SecurityData
{SecurityData optionally appears in the prefix preceding the first segment of all messages.}
+UserDataContainer
+UserDataComponent
0..1
UserData
{UserData optionally appears in the prefix preceding the first segment of all messages.}
0..1
StandardFields
Length : TwoByteField
ReservedField : TwoByteField
TransactionCode : VaiableLengthField
{The Transaction Code field can be from 1 to 8 bytes in length. It's included only in input messages.}
Format "one": a prefix appears before all message segments.
Format "two": a prefix appears only before the first segment.
In direct XML-to-IMS application scenarios. StandardFields are not included.

FIG. 9

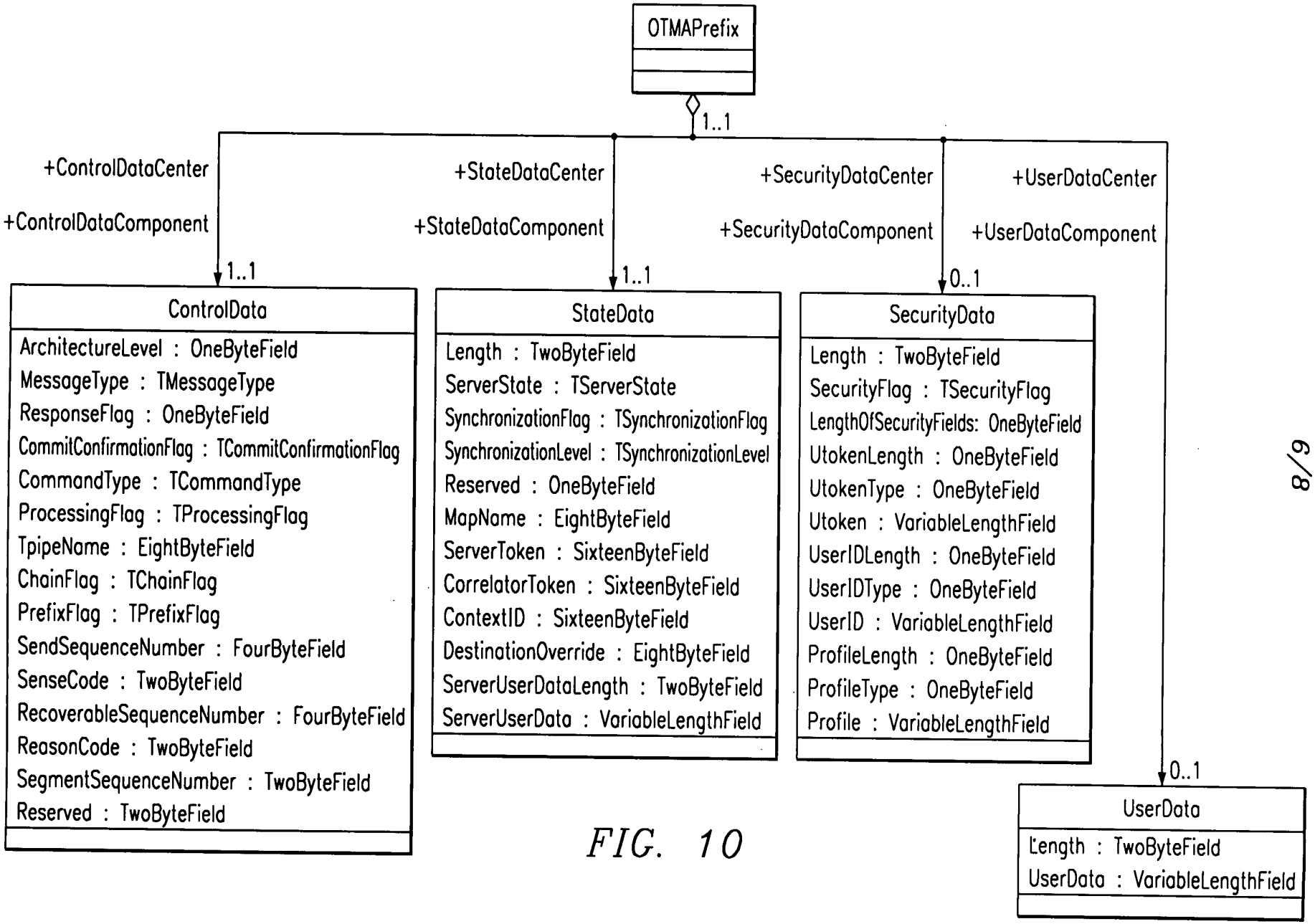
OTMAPrefix
1..1
+ControlDataCenter
+ControlDataComponent
+StateDataCenter
+StateDataComponent
+SecurityDataCenter
+SecurityDataComponent
+UserDataCenter
+UserDataComponent
1..1
1..1
0..1
0..1
ControlData
ArchitectureLevel : OneByteField
MessageType : TMessageType
ResponseFlag : OneByteField
CommitConfirmationFlag : TCommitConfirmationFlag
CommandType : TCommandType
ProcessingFlag : TProcessingFlag
TpipeName : EightByteField
ChainFlag : TChainFlag
PrefixFlag : TPrefixFlag
SendSequenceNumber : FourByteField
SenseCode : TwoByteField
RecoverableSequenceNumber : FourByteField
ReasonCode : TwoByteField
SegmentSequenceNumber : TwoByteField
Reserved : TwoByteField
StateData
Length : TwoByteField
ServerState : TServerState
SynchronizationFlag : TSynchronizationFlag
SynchronizationLevel : TSynchronizationLevel
Reserved : OneByteField
MapName : EightByteField
ServerToken : SixteenByteField
CorrelatorToken : SixteenByteField
ContextID : SixteenByteField
DestinationOverride : EightByteField
ServerUserDataLength : TwoByteField
ServerUserData : VariableLengthField
SecurityData
Length : TwoByteField
SecurityFlag : TSecurityFlag
LengthOfSecurityFields: OneByteField
UtokenLength : OneByteField
UtokenType : OneByteField
Utoken : VariableLengthField
UserIDLength : OneByteField
UserIDType : OneByteField
UserID : VariableLengthField
ProfileLength : OneByteField
ProfileType : OneByteField
Profile : VariableLengthField
UserData
Length : TwoByteField
UserData : VariableLengthField

FIG. 10

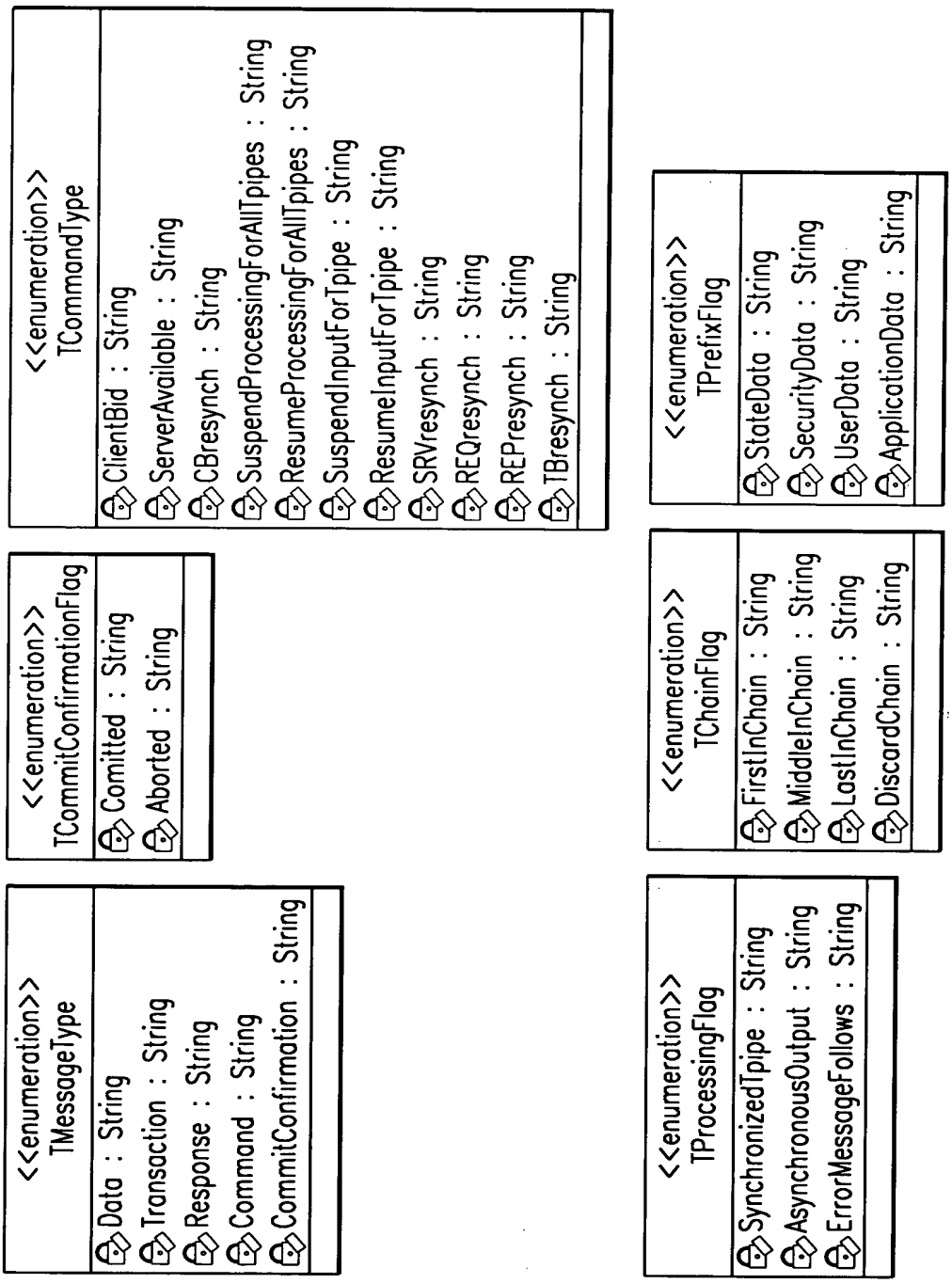
<<enumeration>>
TMessageType
Data : String
Transaction : String
Response : String
Command : String
CommitConfirmation : String
<<enumeration>>
TCommitConfirmationFlag
Comitted : String
Aborted : String
<<enumeration>>
TCommandType
ClientBid : String
ServerAvailable : String
CBresynch : String
SuspendProcessingForAllTpipes : String
ResumeProcessingForAllTpipes : String
SuspendInputForTpipe : String
ResumeInputForTpipe : String
SRVresynch : String
REQresynch : String
REPresynch : String
TBresynch : String
<<enumeration>>
TProcessingFlag
SynchronizedTpipe : String
AsynchronousOutput : String
ErrorMessageFollows : String
<<enumeration>>
TChainFlag
FirstInChain : String
MiddleInChain : String
LastInChain : String
DiscardChain : String
<<enumeration>>
TPrefixFlag
StateData : String
SecurityData : String
UserData : String
ApplicationData : String

FIG. 11

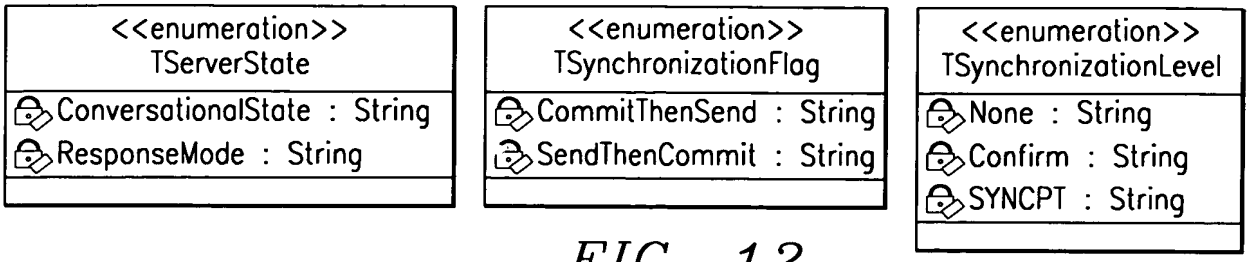

FIG. 12

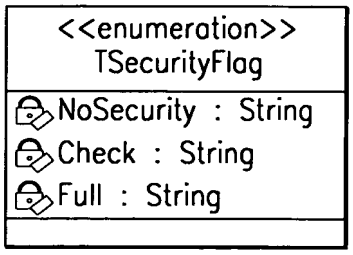

FIG. 13

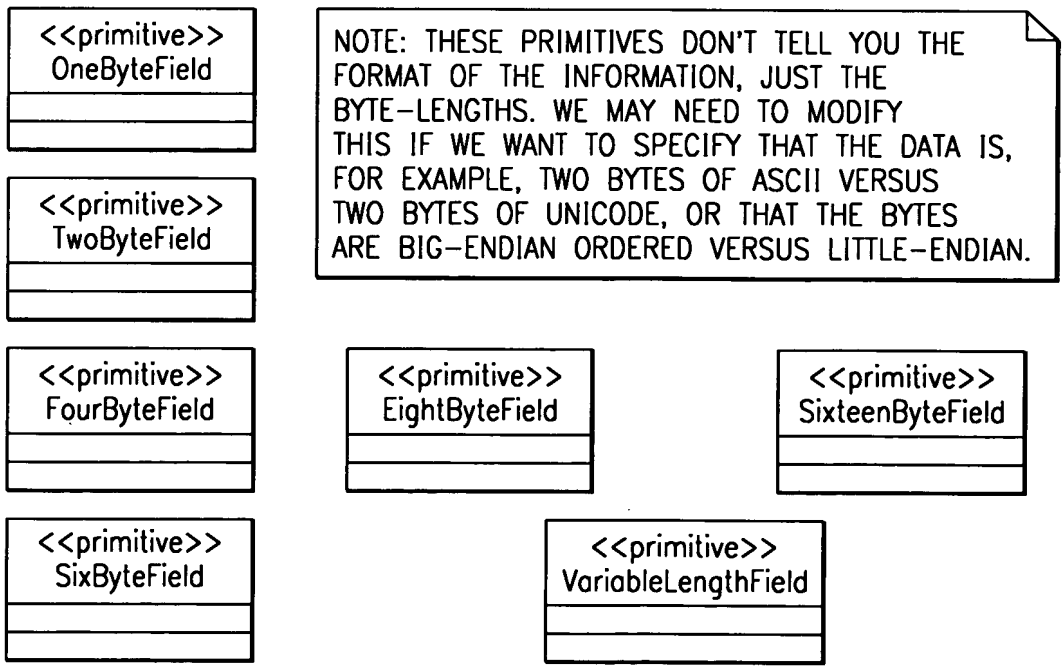

FIG. 14